PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ

DOMINUS

CURITIBA

2012

LUIZ AUGUSTO DE PAULA

RODRIGO BUSATO

DOMINUS

Projeto apresentado a disciplina de Resolução de Problemas em Engenharia I do Curso de Engenharia de Computação da Escola Politécnica sob orientação do Prof. Afonso Ferreira Miguel.

CURITIBA

RESUMO

Projeto para o curso de Engenharia de Computação, no intuito de aprender a projetar e construir um projeto para aprendizado. Projetado uma porta automática com acionamento pelo controle remoto de televisores comuns utilizando o micro controlador Arduino.

ABSTRACT

Project for a class of Computer Engineer, to learn and understand how is to create a project and make one. It is a door who open and close automatic with a remote control, using Arduino.

SUMÁRIO

INTRODUÇÃO

Idosos, cadeirantes ou pessoas com deficiências motoras geralmente tem grandes dificuldades para abrirem ou fecharem as portas de sua residência.

Pensando nisso, surge a ideia de construir um sistema de abertura e fechamento de portas para facilitar o acesso, permitindo economia de tempo e trabalho.

Atualmente existem portas automáticas apenas de correr em trilhos.

OBJETIVOS

Geral

Projetar um sistema para que uma porta abra, feche e trave.

Específicos

Controlar o sentido de rotação do motor utilizado através da ponte H.

Controlar o acionamento da trava eletromagnética para destravar a porta.

Implementar software que comanda o acionamento de todos os componentes.

DESCRIÇÃO GERAL

Projeto

Construir uma porta automática com acionamento por controle remoto de televisão e utilizando o micro controlador Arduino. A porta abre com o acionamento do botão e espera um tempo programável ou se acionado o controle, a porta em estado de espera e fecha com o acionamento novamente do controle.

Quando esta fechando, tem um sensor de movimento, que se for acionado, a porta para imediatamente.

História do Projeto

Ao construir a maquete a madeira utilizada inchava dependendo do clima do da semana fazendo com que sempre houvesse a necessidade de reajustar as dimensões da estrutura que comportava a porta.

No início da implementação do software, o primeiro problema encontrado foi na leitura do sinal do receptor, em que o sinal lido não podia ser comparado. Após muitas pesquisas na internet, encontramos um programa que converte o sinal lido para ser comparado e utilizado no acionamento do sistema.

Ao fazer a integração da parte eletrônica com a mecânica, para movimento do motor, foi verificado que o motor precisaria de 4A para bom funcionamento, logo o recurso utilizando foi a fonte do laboratório ligada em paralelo.

Para o motor girar nos dois sentidos, os CI L293D e L298, não foram suficientes, logo foi necessário construir a ponte H com transistores e reles. Para facilitar na construção da ponte H utilizamos uma placa perfurada própria, onde quatro reles são necessários para a o motor e o último para acionamento da tranca.

Outro problema encontrado foi na utilização de um LED emissor para o sensor de movimento. Como na maquete o sensor não funcionou devidamente, por ficar mais de 30cm do receptor o sinal se tornou insuficiente, então em vez de usar o LED, optamos por utilizar um lazer point.

Foram utilizados botões push bottom servem como sensores fim de curso para limitar o movimento da porta.

Descrição detalhada

Semana de 02/04/2012 à 08/04/2012

Busca de tecnologia disponível, criação de diagrama mecânico e fluxograma de funcionamento da porta.

Semana de 09/04/2012 à 15/04/2012

Desenho da maquete; escolha e aquisição dos componentes eletrônicos; compra do Arduino.

Semana de 16/04/2012 à 22/04/2012

Compra dos elementos da maquete e parte da implementação lógica do software;

Semana de 23/04/2012 à 29/04/2012

Estrutura da maquete pronta exceto colocação do motor e continuação da implementação logica do software;

Semana de 30/04/2012 à 06/05/2012

Conclusão da maquete e implementação do código para leitura dos dados de entrada do sensor.

Semana de 07/05/2012 à 13/05/2012

Encontrar solução para abertura da porta, chegada do Arduino e desenho do shield para acoplar no mesmo.

Semana de 14/05/2012 à 20/05/2012

Acoplar o motor a maquete e construção do shield para o Arduino.

Semana de 21/05/2012 à 27/05/2012

Conclusão da estruturação mecânica e integração da parte mecânica com elétrica.

Semana de 28/05/2012 à 03/06/2012

Coleta de dados obtidos durante o desenvolvimento do projeto para estruturação da documentação do projeto e teste de funcionamento completo;

Semana 03/06/2012 à 10/06/2012

Conclusão da documentação do projeto, ajustes finais e testes finais.

Semana 11/06/2012 à 13/06/2012

Apresentação final e entrega da documentação e vídeo.

MATERIAIS UTILIZADOS

Arduino;

Resistores de fio de 47 Ω, 39 Ω, 10k Ω, 4700 Ω;

Capacitor de 47µF 63V;

Push bottom;

LED IR;

Motor DC 12V 4A;

Trava eletromecânica 12V AC;

Fechadura;

Dobradiças;

Molas de pressão;

Transistor 222A;

Rele 5 pinos 5V;

Placa de protótipo rele;

Placa perfurada;

Caixa patola PB 119;

Fios;

Laser point;

Madeira;

Parafusos;

Pinos poste;

Termo retrátil.

PROBLEMAS APRESENTADOS

PROBLEMAS	SOLUÇÃO
Implementação do motor com mecanismos de movimentação da porta (coroa e rosca sem fim) não encaixam em determinados ângulos de abertura da porta, devido as dilatações sofridas pela madeira utilizada e ao uso de dobradiças com baixa precisão.	Uso de cabo para tracionar a porta para realizar os movimentos.
Abertura da porta através do cabo se torna inviável, pois o mesmo tinha finalidade apenas para fechar.	Uso de molas acopladas próximas as dobradiças tornado a abertura espontânea assim que a trava era aberta, com isso o uso do cabo permaneceu.
Fazer o motor utilizado trabalhar nos dois sentidos de rotação.	Uso do LM298, entretanto, não eficaz. Foi desenvolvido uma ponte H para acionar o motor.
Uso de contrapeso para ajudar o movimento de abertura e fechamento da porta.	Abandonada a ideia, pois no protótipo não influenciava demais elementos.
Posicionamento do motor causava uma perda para o movimento de abertura da porta	Uso de outra mola e mudança de posição.
Interpretação do sinal do receptor infravermelho.	Uso de software pronto pesquisado na internet.

MANUAL DO USUÁRIO

A utilização da porta automática é muito simples, aonde o usuário deve somente memorizar dois comandos, botão 1 e botão 2.

Para abertura inicial da porta, o usuário deve pressionar o botão 1, se deseja que a porta feche automaticamente, não necessita de nenhum acionamento, ira fechar em 30 segundos ou o tempo programado para executar o fechamento.

Ao termino da abertura e durante os 30 segundos para fechar, se pressionado o botão 2, entrara em estado de espera, aonde fica aberta até acionamento novamente do botão 1.

Com um sensor de movimento na porta, ao ser acionado, para imediatamente e só volta a funcionar ao sensor de movimento ser liberado.

MANUAL DE INSTALAÇÃO

Para instalação o usuário deve primeiramente ter a porta estalada em sua casa. Ter um espaço de sobra em cima da porta para adicionar o motor, ponte-h e o Arduino, no trinco da porta, local para por a tranca.

Instalar um push boton onde a porta encontra quando fecha e quando abre, para leitura do sinal.

Ter um local na lateral da porta para inserção do led receptor, e outro na outra extremidade, na mesma direção, para instalar o lazer point.

Conectar todos os fios no shield, ponte-h conforme desenho nos anexos.

Alinhar o lazer point exatamente no led receptor e testar a porta.

CÓDIGO FONTE

```
#include <IRremote.h>

int sen_fim = 7;

int sen_abre = 8;
int mot_abre = 9;
int mot_fecha = 10;
int tranca = 11;
int sen_mov = A7;

unsigned static long hash;

  #define FNV_PRIME_32 16777619
  #define FNV_BASIS_32 2166136261


  int RECV_PIN = 6;    //Onde será ligado o Receptor IR
  IRrecv irrecv(RECV_PIN);
  decode_results results;

  void setup()
  {

    pinMode(tranca, OUTPUT); //tranca

    pinMode(mot_abre, OUTPUT); //motor abre
    pinMode(mot_fecha, OUTPUT); // motor fecha
    pinMode(sen_mov, INPUT); // sensor movimento
    pinMode(sen_fim, INPUT); // sensor fim de curso
    pinMode(sen_abre, INPUT); // sensor de abrir


    irrecv.enableIRIn();   //Iniciando a recepção do sinal
    Serial.begin(9600);
  }

    int compare(unsigned int oldval, unsigned int newval) {
    if (newval < oldval * .8) {
      return 0;
    }
    else if (oldval < newval * .8) {
      return 2;
    }
    else {
      return 1;
    }
```

```
}

// compare (120, 230);

unsigned long decodeHash(decode_results *results) {
  unsigned long hash = FNV_BASIS_32;
  for (int i = 1; i+2 < results->rawlen; i++) {
    int value =  compare(results->rawbuf[i], results->rawbuf[i+2]);
    hash = (hash * FNV_PRIME_32) ^ value;
  }
  return hash;
}

void decodifica() {
    if (irrecv.decode(&results)) {
    Serial.print("'real' decode: ");     //Mostra o valor recebido
    Serial.print(results.value, HEX);    //!!!!!!!PARA IDENTIFICAR AS TEClAS
    Serial.print(", hash decode: ");
    hash = decodeHash(&results);
   Serial.println (hash);                //Mostra o valor já decodificado!
    irrecv.resume(); // Obrigatório a reinicialização da recepção!
  }
}

void loop() {


  decodeHash(&results);
  decodifica();


//-------------Local para inserção do código logo abaixo!----------------
int i, j = 1, k = 1,p = 1,l = 1,val= 0;


 //

  decodeHash(&results);
  decodifica();
  if (hash == 1458516811 || hash == 3268448703) { // verifica botao apertado
   Serial.println(results.value, HEX);

   irrecv.resume(); // Receive the next value

   Serial.println("recebeu");
   irrecv.resume(); // recebe o procimo valor
     digitalWrite(tranca, HIGH); // rele tranca
     delay(15);
```

```
digitalWrite(tranca, LOW

); // rele tranca

Serial.println("tranca");
digitalWrite(mot_abre, HIGH); //  rele aciona motor

Serial.println("motor abre");
do{
  Serial.println("LENDO SENSOR comeco");
    val = digitalRead(sen_abre);  // ler o valor de entrada
      if (val == LOW) {         // verificar se a entrada é HIGH (interruptor livre)
                  digitalWrite(mot_abre, LOW); //rele desliga motor
                  Serial.println("desliga motor começo");
      }
}while(val == HIGH);

irrecv.resume(); // recebe o procimo valor
for(int i = 0; i<= 3000; i++){

  decodeHash(&results);
  decodifica();
  Serial.println("dentro do for");
  Serial.println(results.value, HEX);

  if (hash == 389103054 || hash == 1384796706
 ){// pede para parar

        Serial.println("recebeu parou");

          do{
            decodeHash(&results);


            decodifica();
            if(hash == 1458516811 || hash == 3268448703){ // recebe para voltar a
funcionar

              Serial.println("recebeu pra funciona denovo");
              digitalWrite(mot_fecha, HIGH);// aciona motor fechar
              Serial.println("motor fecha dentro do for");

              do{
               val = analogRead  (sen_mov);  // ler o valor de entrada
               if(val == LOW ) {// sensor de movimento
                      digitalWrite(mot_fecha, LOW);//para motor
                      Serial.println("sensor movimento acionado   ");
               }else{
```

```
                digitalWrite(mot_fecha, HIGH);// funciona o motor
                Serial.println("motor fecha");
                    Serial.println("LENDO SENSOR FIM DE CURSO for");
                      val = digitalRead(sen_fim);  // ler o valor de entrada
                        if (val == LOW      ) {         // verificar se a entrada é HIGH
(interruptor livre)
                            digitalWrite(mot_fecha, LOW);//para motor
                            digitalWrite(tranca, LOW);//tranca porta
                            Serial.println(" sensor fim do curso tranca porta");
                            k = 0;
                            i = 3001;

                            j = 0;
                            irrecv.resume(); // recebe o procimo valor


                        }
              }


            }while(k == 1);
            }}  while (j == 1);
    }else{
      i = i+ 300;
      delay(1000);
    }

  }
  val = digitalRead(sen_fim);  // ler o valor de entrada


  if (val == HIGH ) { //verifica se porta fechada
    do{
          val = analogRead  (sen_mov);  // ler o valor de entrada
          if(val == LOW ) {// sensor de movimento
            digitalWrite(mot_fecha, LOW);//para motor
            Serial.println("sensor movimento acionado ");

          }else{
            digitalWrite(mot_fecha, HIGH);// aciona motor
            Serial.println("LENDO SENSOR FIM DE CURSO");
            val = digitalRead(sen_fim);  // ler o valor de entrada
            if (val == LOW) {

                // verificar se a entrada é HIGH (interruptor livre)
              digitalWrite(mot_fecha, LOW);
              Serial.println("porta fecha");
              digitalWrite(tranca, LOW);//tranca porta
```

```
                p = 0;
           }
           }

        }while (p == 1);
   }
  }

hash = FNV_BASIS_32;

  }
```

CONCLUSÃO

Ao final do projeto foi possível cumprir com o que foi proposto, a abertura e o fechamento de uma porta.

Apesar do abandono da ideia de usar uma rosca sem fim, coroa e o lastro para a porta funcionar verificou-se que os mesmos não fariam tanta diferença para este protótipo, entretanto em uma aplicação real, as ideias deixadas de lado seriam bastante interessantes.

Durante a execução do projeto foi, muito se aprendeu, principalmente diante dos problemas encontrados, onde foi necessário um grande empenho para contorná-los, assim, agregando muitos conhecimentos que poderão ser empregados em projetos futuros.

Para que o projeto fosse melhor aprimorado, seria bastante interessante a utilização do que foi desenvolvido em uma porta real, pois a miniatura gerou bastantes problemas durante sua confecção.

REFERENCIAS

Botão ARDUINO - http://arduino.cc/playground/Portugues/LearningButton;

Montagem ponte – h - http://www.maxwellbohr.com.br/downloads/robotica/mec1000_kdr5000/tutorial_eletronica_-_montagem_de_uma_ponte_h.pdf;

Receptor IF - http://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CGoQFjAA&url=http%3A%2F%2Fwww.arduino.cc%2Fcgi-bin%2Fyabb2%2FYaBB.pl%3Fnum%3D1278969179&ei=EAvZT-_6BYi68ASDl73oAw&usg=AFQjCNHXLPK513OSB_AQ2qM96xYSpMCH5Q;

LED IF - http://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CHAQFjAA&url=http%3A%2F%2Fforum.clubedohardware.com.br%2Fled-infravermelho-infrared%2F476386&ei=bwvZT6nMJIHs8wSzuLnrAw&usg=AFQjCNFy_u4Qoj7Qa1uN7aFVQi4LqNzf1g

ANEXOS
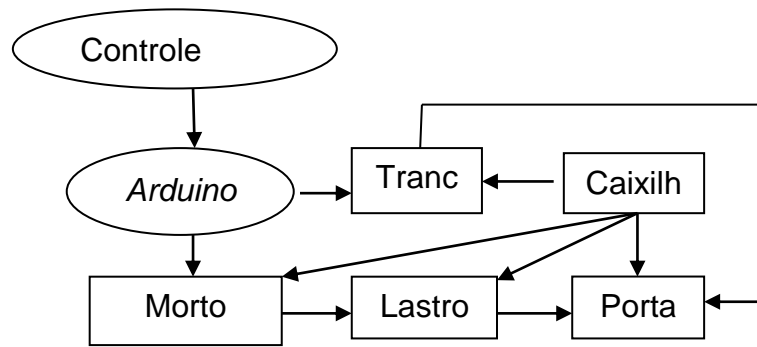
Figura 1: Diagrama de funcionamento mecânico;
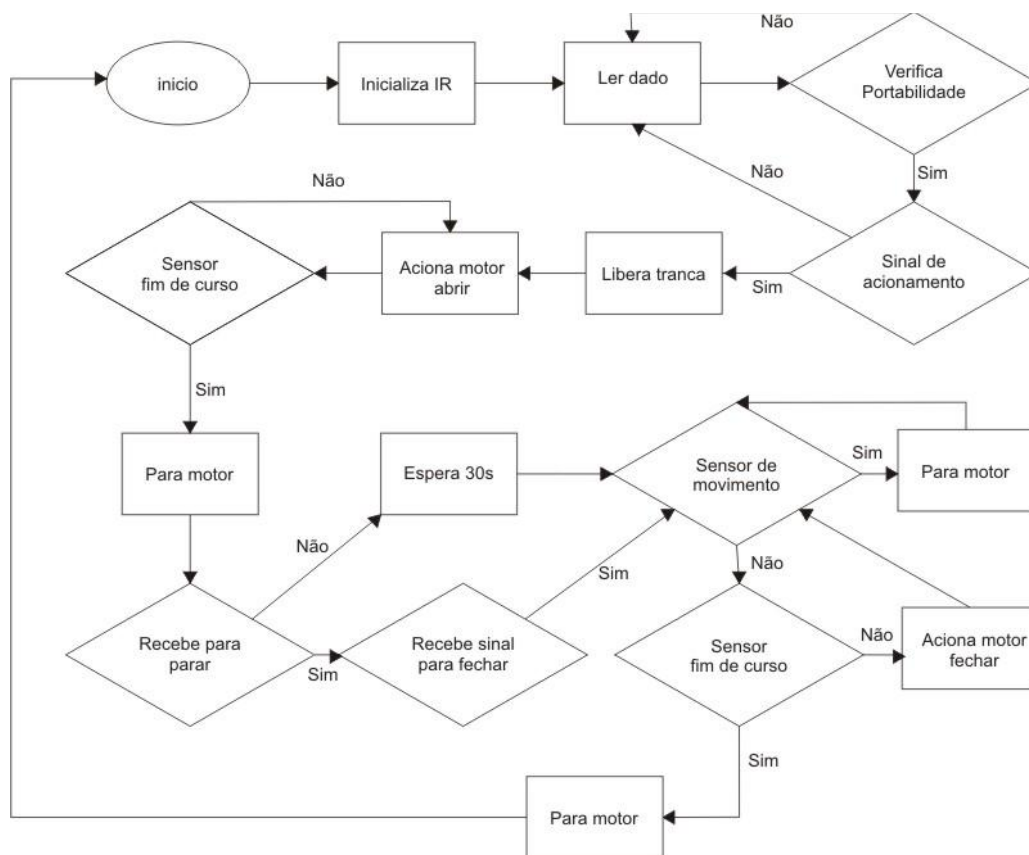
Figura 2: Diagrama de funcionamento do software;
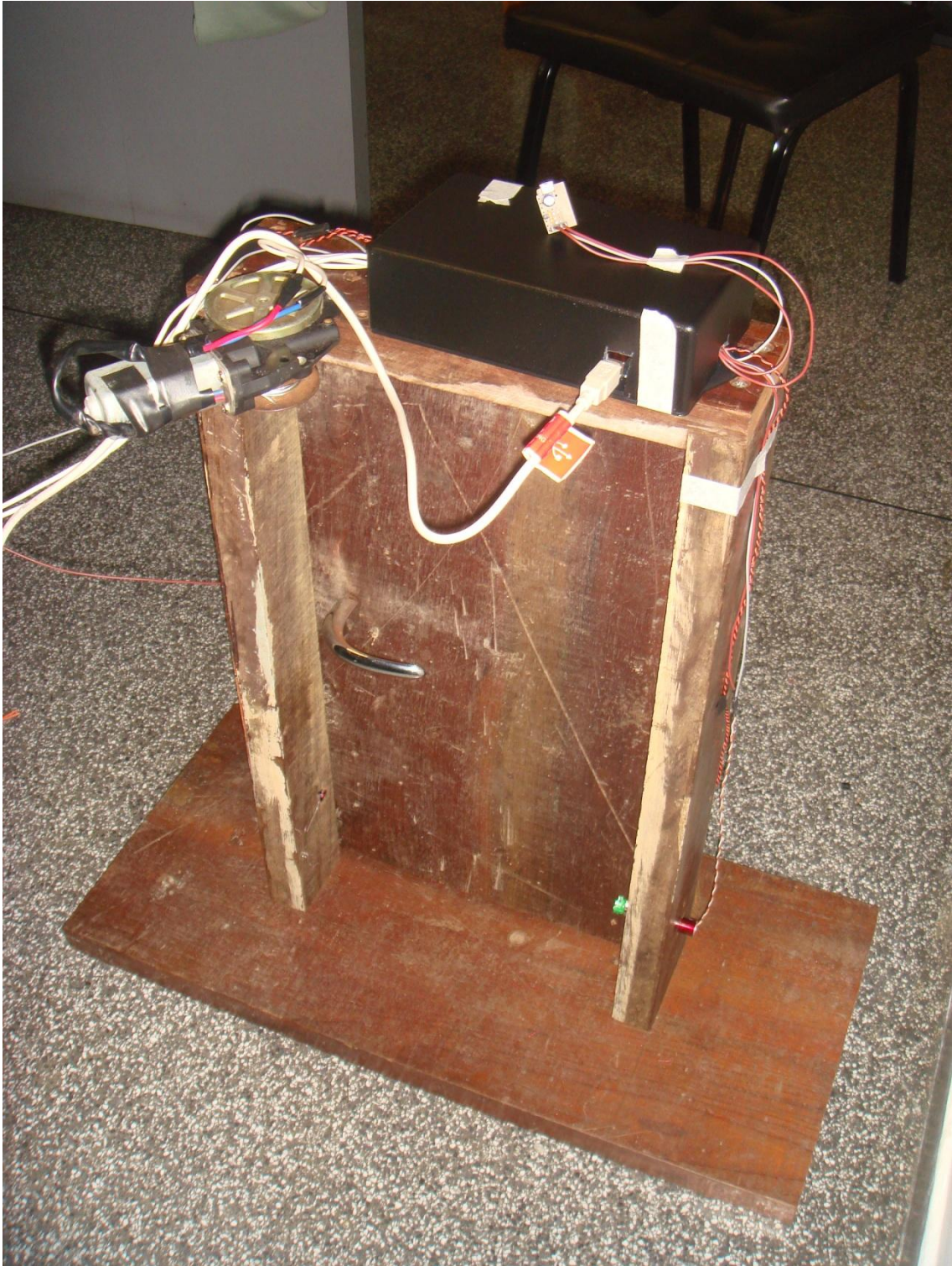
Figura 3: Maquete montada;
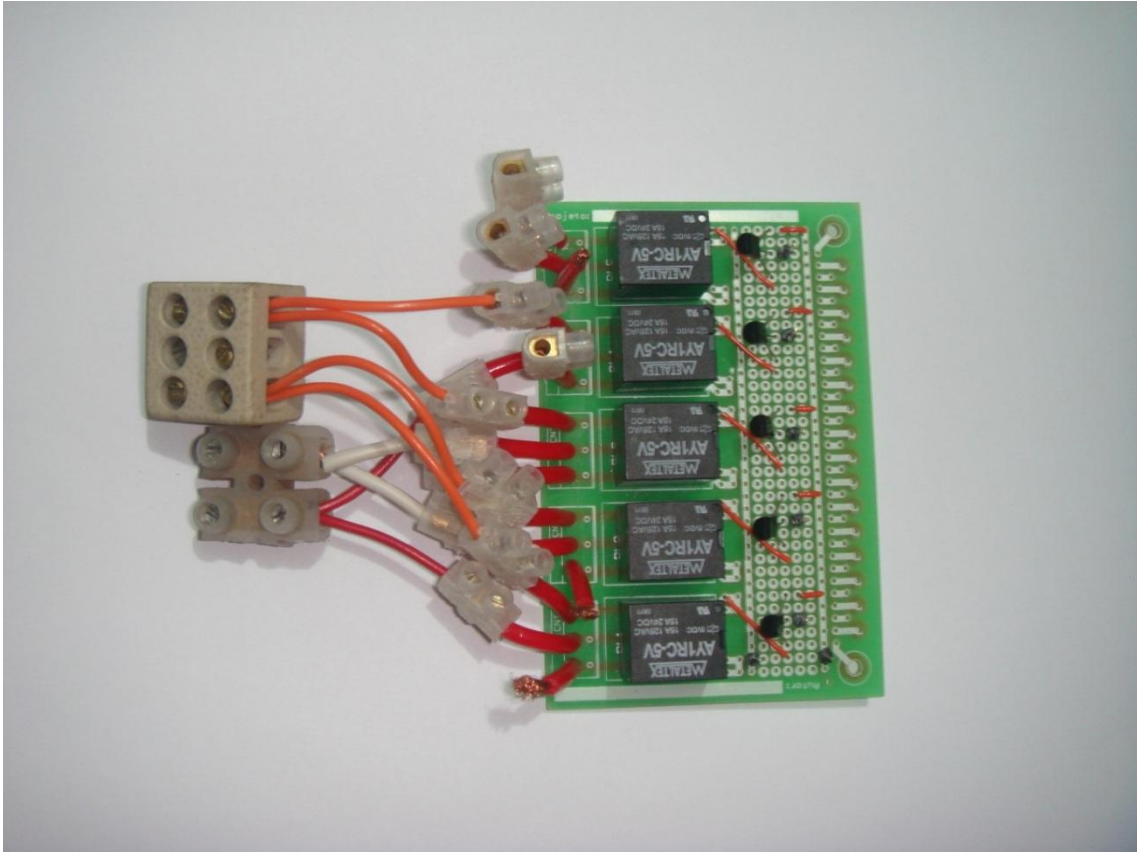
Figura 4: Ponte H desenvolvida;
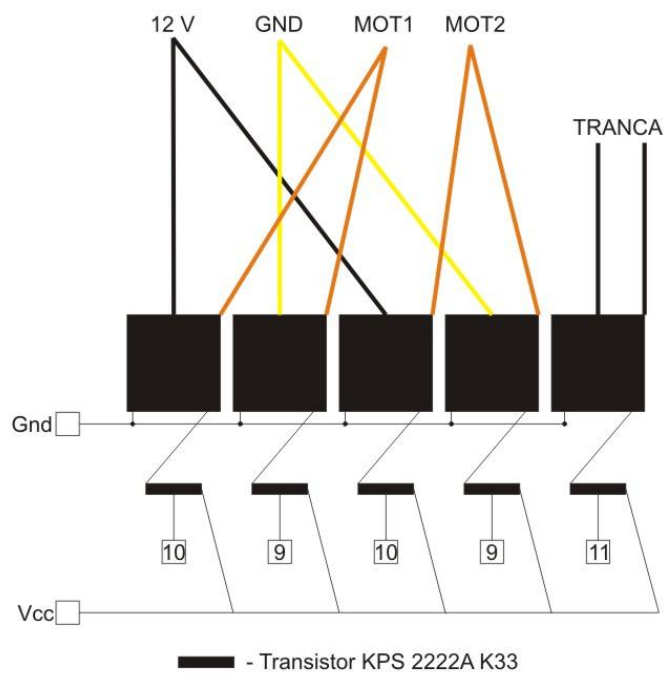
Figura 5: Desenho ponte-h;
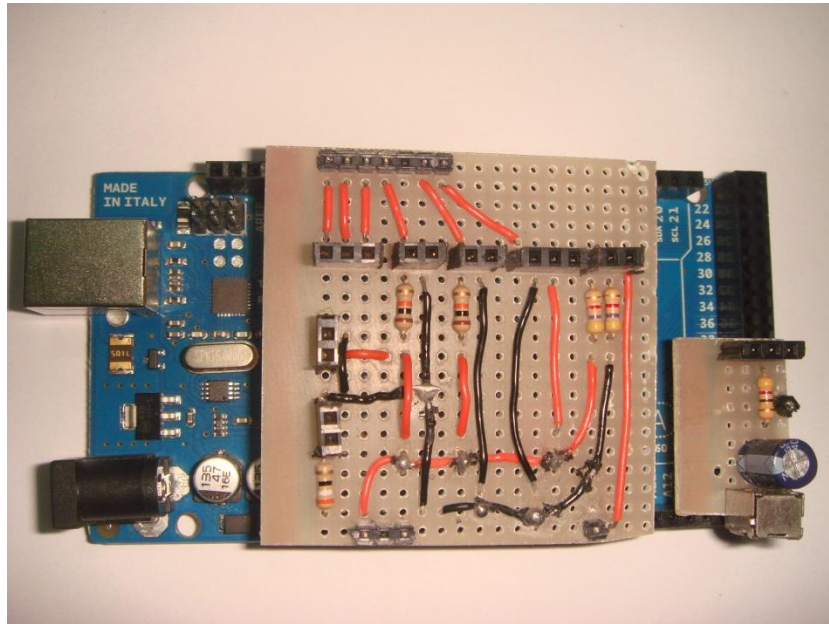
Figura 6: Shield sobre o Arduino;
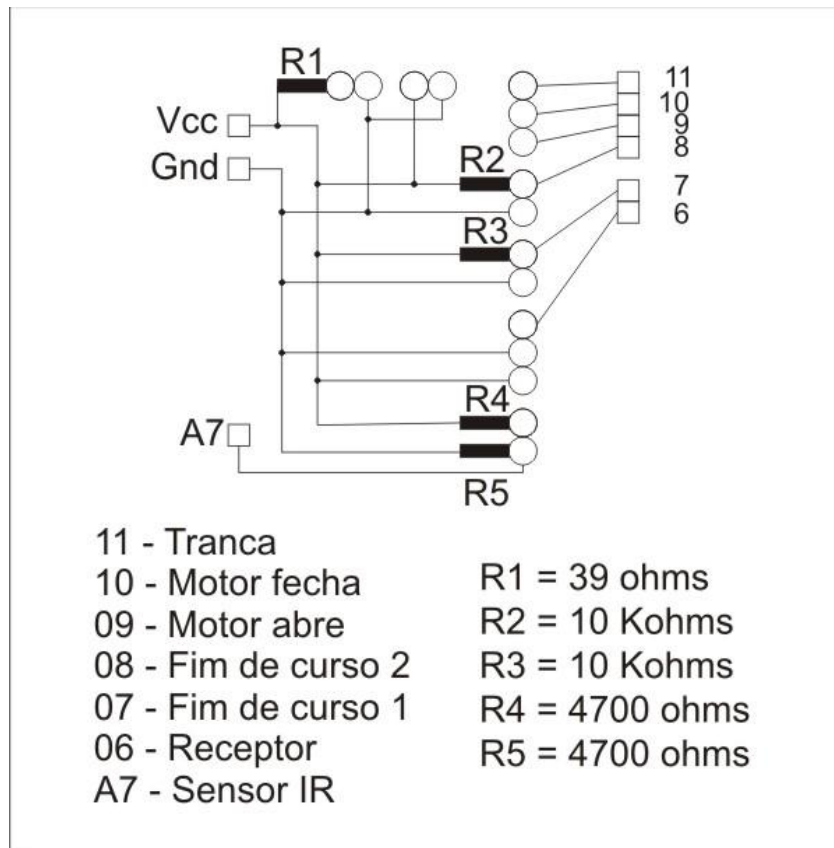
Figura 7: Shield Arduino;

Figura 8 Laser point